

Строки

Что такое строка?

- Последовательность символов, заключенная в кавычки
- `line = "текстовая строка"`
- Виды кавычек:
 - ' # одинарные
 - " # двойные
 - ''' """ # тройные (мультистрочный текст)
- Строки могут содержать буквы, цифры, пробелы и другие символы.

Строки неизменяемы

- `s = "Hello"` # начальная строка
 `s = "Hi"` # создаем новую строку
 `s = s[1:]` # создаем новую строку из начальной
 `s[0] = "h"` # ошибка!
- `x += 1` # создаем новую строку
 `x = x + 1` # эквивалент

Соединение строк

- `meet = "баранина"`
`fruit = "банан"`

```
dinner = meet + " и " + fruit    # конкатенация строк  
dinner = f"{meet} {fruit}"      # форматированная строка
```

```
meet_amount = meet * 3          # дублируем строку
```

```
greeting = "Порций {}: {}".format(meet, meet_amount)
```


Кодировки

- **ASCII** (128 символов, 1 байт). Латиница, цифры, спецсимволы и управляющие символы
- **UTF-8**: (1 114 112 символов, до 4 байта). Все языки мира*, музыкальные ноты, эмодзи, графические символы.
- **UNICODE**: (1 357 024 символов)
- **utf8** - кодировка стандарта **unicode**

Содержание строк

- Обычные строки - содержание вначале парсится:

```
sample = "Hi 😊\n12345 \t спецсимволы: \" \\ / () " +  
         "`~\n Unicode: \u2600\u2601\u2602"
```

- | | |
|-------------------------------------|------------------------------|
| <code>\n</code> | # новая строка |
| <code>\t</code> | # табуляция |
| <code>\"</code> или <code>\\</code> | # экранирование спец символа |
| <code>\u1234</code> | # символ unicode кодировки |

- "Сырые" строки - символы учитываются буквально, без парсинга:
`raw_str = r"this\nis\tone\nline\u2602"`

Срез строки

- `string[start:end:step]`
- `line = "123456789"`
 - `v1 = line[3]` # 4
 - `v1 = line[5:]` # 6789
 - `v1 = line[6::]` # 789
 - `v2 = line[:4:]` # 1234
 - `v3 = line[::2]` # 13579
 - `v4 = line[:9:3]` # 147
 - `v5 = line[0:6:2]` # 135
 - `v6 = line[-2::]` # 89
 - `v7 = line[-3::-2]` # 7531

Методы строк

Самые популярные методы (всего их в Python - 47)

- `s = " Привет! "`
`s.lower()` # " привет! "
`s.upper()` # " ПРИВЕТ! "
`s.strip()` # "Привет!"
`s.replace("иве", "ive")` # " Прівет! "
`s.split("и")` # [' Пр', 'вет! ']
`"-".join(["1", "2", "3"])` # 1-2-3
`s.find("иве")` # 3
`s.count(" ")` # 2
`s.startswith(" При")` # True

Модуль `string`

содержит много полезного

- Импорт модуля:
`import string`

- Форматирование строк:
`from string import Template`
`template = Template("Привет, $name! Заходи в $place.")`
`result = template.substitute(place="кафе", name="Тэд")`
`print(result) # Привет, Тэд! Заходи в кафе.`